

Title of the Prior Art

Japanese Published Patent Application No. 2000-050269

Date of Publication : February 18, 2000

Inventors : M. HYODO et al.

Concise Statement of Relevancy

This reference discloses an apparatus for concealing an error in moving picture conversion coding, in which concealment processing is performed to data including an error which is detected by error correcting decoding and cannot be corrected, with using data of a pixel at a position which is temporally and spatially optimum dependent on the type of the data.

THIS PAGE BLANK (USPTO)

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-50269

(P2000-50269A)

(43) 公開日 平成12年2月18日 (2000.2.18)

(51) Int.Cl.⁷

H 0 4 N 7/30

識別記号

F I

H 0 4 N 7/133

マークシート (参考)

A

審査請求 有 請求項の数 6 O L (全 18 頁)

(21) 出願番号 特願平11-192457
(62) 分割の表示 特願平3-320461の分割
(22) 出願日 平成3年12月4日 (1991.12.4)

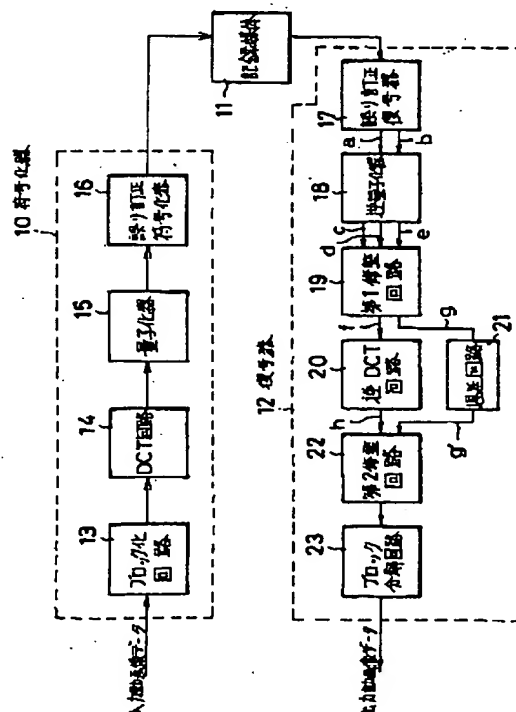
(71) 出願人 000005049
シャープ株式会社
大阪府大阪市阿倍野区長池町22番22号
(72) 発明者 兵頭 正晃
大阪府大阪市阿倍野区長池町22番22号 シ
ャープ株式会社内
(72) 発明者 堅田 裕之
大阪府大阪市阿倍野区長池町22番22号 シ
ャープ株式会社内
(72) 発明者 野口 要治
大阪府大阪市阿倍野区長池町22番22号 シ
ャープ株式会社内
(74) 代理人 100103296
弁理士 小池 隆彌

(54) 【発明の名称】 復号装置

(57) 【要約】

【課題】 誤り訂正復号処理において、従来から行われている、周辺の画素値を用いる修整方法では、誤りの発生した画素の周辺の画素値が正常に再生されている場合には有効であるが、誤りの生じている画素の周辺画素に誤りが生じている場合には有効ではない。また、例えば、画像全体に大きな影響を与える付加情報等に誤りがあった場合に、有効ではない。

【解決手段】 前記誤り訂正復号手段において、誤り検出はできたが、誤り訂正できなかった復号データが、付加情報であるか否かを判断する判断手段と、前記判断手段において、付加情報と判断された場合に、当該ブロックにおけるすべての再生画素値を、前画面における当該ブロックと同位置のブロックの再生画素値に置き換える修整手段とを備えることを特徴とする。



【特許請求の範囲】

【請求項 1】 ブロック毎の画素値を直交変換して求められる変換係数を、付加情報に基づいて量子化した量子化インデックスと、前記付加情報とを含むデータを符号化した符号化データを入力して誤り訂正復号を行い、復号データを出力する誤り訂正復号手段と、前記誤り訂正復号手段において、誤り検出はできたが、誤り訂正できなかった復号データが、付加情報であるか否かを判断する判断手段と、前記判断手段において、付加情報と判断された場合に、当該ブロックにおけるすべての再生画素値を、前画面における当該ブロックと同位置のブロックの再生画素値に置き換える修整手段とを備えることを特徴とする復号装置。

【請求項 2】 ブロック毎の画素値を直交変換して求められる変換係数を、付加情報に基づいて量子化した量子化インデックスと、前記付加情報とを含むデータを符号化した符号化データを入力して誤り訂正復号を行い、復号データを出力する誤り訂正復号手段と、前記誤り訂正復号手段において、誤り検出はできたが、誤り訂正できなかった復号データが、付加情報であるか否かを判断する判断手段と、前記判断手段において、付加情報と判断された場合に、当該ブロック中のすべての変換係数を、前画面における当該ブロックと同位置のブロックの変換係数に置き換える修整手段とを備えることを特徴とする復号装置。

【請求項 3】 ブロック毎の画素値を直交変換して求められる変換係数を量子化した量子化インデックスを含むデータを符号化した符号化データを入力して誤り訂正復号を行い、復号データを出力する誤り訂正復号手段と、前記誤り訂正復号手段において、誤り検出はできたが、誤り訂正できなかった復号データが、量子化インデックス中の低周波成分であるか否かを判断する判断手段と、前記判断手段において、低周波成分と判断された場合に、当該ブロックにおけるすべての再生画素値を、前画面における当該ブロックと同位置のブロックの再生画素値に置き換える修整手段とを備えることを特徴とする復号装置。

【請求項 4】 ブロック毎の画素値を直交変換して求められる変換係数を量子化した量子化インデックスを含むデータを符号化した符号化データを入力して誤り訂正復号を行い、復号データを出力する誤り訂正復号手段と、前記誤り訂正復号手段において、誤り検出はできたが、誤り訂正できなかった復号データが、量子化インデックス中の低周波成分であるか否かを判断する判断手段と、前記判断手段において、低周波成分と判断された場合に、当該ブロック中のすべての変換係数を、前画面における当該ブロックと同位置のブロックの変換係数に置き換える修整手段とを備えることを特徴とする復号装置。

【請求項 5】 ブロック毎の画素値を直交変換して求められる変換係数を量子化した量子化インデックスを含む

データを符号化した符号化データを入力して誤り訂正復号を行い、復号データを出力する誤り訂正復号手段と、前記誤り訂正復号手段において、誤り検出はできたが、誤り訂正できなかった復号データが、量子化インデックス中の高周波成分であるか否かを判断する判断手段と、前記判断手段において、高周波成分と判断された場合に、誤り訂正できなかった復号データに対応する変換係数のみを 0 で置き換える修整手段とを備えることを特徴とする復号装置。

【請求項 6】 ブロック毎の画素値を直交変換して求められる変換係数を量子化した量子化インデックスを含むデータを符号化した符号化データを入力して誤り訂正復号を行い、復号データを出力する誤り訂正復号手段と、前記誤り訂正復号手段において、誤り検出はできたが、誤り訂正できなかった復号データが、水平方向に隣接するブロックと相関の高い周波数成分、或いは、垂直方向に隣接するブロックと相関の高い周波数成分であるか否かを判断する判断手段と、前記判断手段において、水平方向に隣接するブロックと相関の高い周波数成分と判断された場合には、誤り訂正できなかった復号データに対応する変換係数のみを、水平方向に隣接するブロックの同一周波数成分の変換係数で置き換え、垂直方向に隣接するブロックと相関の高い周波数成分と判断された場合には、誤り訂正できなかった復号データに対応する変換係数のみを、垂直方向に隣接するブロックの同一周波数成分の変換係数で置き換える修整手段と、を備えることを特徴とする復号装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、符号化された画像データを復号するときに読み出した画像データの誤りを修整する復号装置に関する。

【0002】

【従来の技術】一般に、効率よくデジタル動画データ等を磁気テープや磁気ディスク等に記録したり、回線を用いて伝送する場合には、記録する情報量を少なくするために画像データを高能率で符号化することが知られている。

【0003】動画データを高能率で符号化する方法の 1 つとしては、一部が本発明と同一の発明者による特願平 3-118825 号に記載の方法が提案されている。これは 1 画面のデータを例えば 8 画素×8 画素のブロックに分解し、ブロック毎に直交変換を行い、変換係数をブロックの統計的性質に基づいて適応的に量子化する方法である。

【0004】一般に、量子化を行った後、量子化データ及びそれに付随する付加情報は誤り訂正符号化され、誤り訂正符号化された量子化データや付加情報は記録媒体に記録される。

【0005】上記の方法で記録されたデータを読み出す

ときに、値が誤って読み出されることが起こり得る。誤り訂正符号化が行われている符号の場合、読み出したデータには誤り訂正復号が行われ、誤ったデータのうち幾つかのデータは誤りの生じた位置がわかる誤り検出が行われ、更に、幾つかのデータは正しい値に訂正できる誤り訂正が行われる。

【0006】しかし、誤り検出はできたが、誤り訂正ができなかったデータに関しては、そのデータの値が誤っていることはわかっているが、正しい値はわかっていない。そして、誤ったデータのままで復号を行うと、大きな画質劣化として再生画像に現れる。

【0007】そこで、誤り検出しかできなかったデータに対して、画像の相関を利用して誤りを目立たなくする誤り修整が行われる。

【0008】第1の従来の誤り修整方法としては、再生画像上で、誤った画素の周辺画素から実際の画素値を予測し、誤ったデータをその予測値で置き換える方法がある。例えば、図20において、a～fは画素値を表し、誤り検出によりeの値が誤りであることがわかっているときには、次式(1)に示すように、周辺画素からeの値を予測して、

$$e' = (b + d) / 2 \quad \dots\dots (1)$$

として、誤っているeの値をe'で置き換えることが行われる。

【0009】また、第2の従来の誤り修整方法としては、特開昭 61-147690号公報に示されているように、画像をブロックに分解し、ブロック毎の符号化出力として得られるブロックの画素値の最小値とブロック内の画素値のダイナミックレンジからなる付加コード、各画素値とブロックの画素値の最小値との差を所定のビット数で量子化した符号化コードのうち、付加コードに誤りがあった場合に、周辺の付加コードの平均値を予測値とし、誤りのあった付加コードの値をその予測値で置き換える方法である。

【0010】

【発明が解決しようとする課題】通常、変換符号化を用いた方法では、ある特定の画素値に誤りが生じるのではなく、変換係数に誤りが生じる。ブロック内の誤った変換係数を含んだまま逆変換を行った場合には、誤りはブロック全体に影響する。従って、再生画像上では、ブロックの画素値が全て誤りとなるが、上述した第1の従来の誤り修整方法では、誤った画素の周辺の画素値から真の値を予測するので、誤りの発生した画素の周辺の画素値が正常に再生されている場合には有効であるが、誤りの生じている画素の周辺画素にも誤りが生じている場合には不適用である。例えば図1でa～fの全ての画素に誤りが生じている場合に、eの値を修正するために周辺の画素から上記式(1)を用いて予測を行うと、予測に用いるb、dの値が間違っているために精度のよい予測ができず、ブロック全体に及ぶ誤りを修整することがで

きないという問題点があった。

【0011】また、上述した第2の従来の誤り修整方法では、ブロックの画素値の最小値やダイナミックレンジといったブロック全体に大きな影響を与える付加情報を隣接ブロックの相関を利用して予測するので、隣接ブロックと誤りが生じたブロックとの相関が大きい場合には有効であるが、相関が小さい場合には、修整をうけるブロック全体の画素値が真の値と大きく異なってしまうという問題がある。

【0012】更に、ブロック全体に大きな影響を与えるデータの場合、予測値が実際の値と少し違うだけで視覚的に大きな劣化として現れ、予測値を平均値としているために画像全体にめりはりがなくなってしまうという問題点があった。

【0013】本発明は、上述した従来の動画像変換符号化における誤り修整方法における問題点に鑑み、変換符号化を用いて符号化したデータについて、変換係数の性質に適しており時間的な相関をも考慮できる動画像変換符号化誤り修整装置を提供する。

【0014】

【課題を解決するための手段】本発明の第1の発明によれば、ブロック毎の画素値を直交変換して求められる変換係数を、付加情報に基づいて量子化した量子化インデックスと、前記付加情報とを含むデータを符号化した符号化データを入力して誤り訂正復号を行い、復号データを出力する誤り訂正復号手段と、前記誤り訂正復号手段において、誤り検出はできたが、誤り訂正できなかった復号データが、付加情報であるか否かを判断する判断手段と、前記判断手段において、付加情報と判断された場合に、当該ブロックにおけるすべての再生画素値を、前画面における当該ブロックと同位置のブロックの再生画素値に置き換える修整手段とを備えることにより、上記課題を解決する。

【0015】本発明の第2の発明によれば、ブロック毎の画素値を直交変換して求められる変換係数を、付加情報に基づいて量子化した量子化インデックスと、前記付加情報とを含むデータを符号化した符号化データを入力して誤り訂正復号を行い、復号データを出力する誤り訂正復号手段と、前記誤り訂正復号手段において、誤り検出はできたが、誤り訂正できなかった復号データが、付加情報であるか否かを判断する判断手段と、前記判断手段において、付加情報と判断された場合に、当該ブロック中のすべての変換係数を、前画面における当該ブロックと同位置のブロックの変換係数に置き換える修整手段とを備えることにより、上記課題を解決する。

【0016】本発明の第3の発明によれば、ブロック毎の画素値を直交変換して求められる変換係数を量子化した量子化インデックスを含むデータを符号化した符号化データを入力して誤り訂正復号を行い、復号データを出力する誤り訂正復号手段と、前記誤り訂正復号手段にお

いて、誤り検出はできたが、誤り訂正できなかった復号データが、量子化インデックス中の低周波成分であるか否かを判断する判断手段と、前記判断手段において、低周波成分と判断された場合に、当該ブロックにおけるすべての再生画素値を、前画面における当該ブロックと同位置のブロックの再生画素値に置き換える修整手段とを備えることにより、上記課題を解決する。

【0017】本発明の第4の発明によれば、ブロック毎の画素値を直交変換して求められる変換係数を量子化した量子化インデックスを含むデータを符号化した符号化データを入力して誤り訂正復号を行い、復号データを出力する誤り訂正復号手段と、前記誤り訂正復号手段において、誤り検出はできたが、誤り訂正できなかった復号データが、量子化インデックス中の低周波成分であるか否かを判断する判断手段と、前記判断手段において、低周波成分と判断された場合に、当該ブロック中のすべての変換係数を、前画面における当該ブロックと同位置のブロックの変換係数に置き換える修整手段とを備えることにより、上記課題を解決する。

【0018】本発明の第5の発明によれば、ブロック毎の画素値を直交変換して求められる変換係数を量子化した量子化インデックスを含むデータを符号化した符号化データを入力して誤り訂正復号を行い、復号データを出力する誤り訂正復号手段と、前記誤り訂正復号手段において、誤り検出はできたが、誤り訂正できなかった復号データが、量子化インデックス中の高周波成分であるか否かを判断する判断手段と、前記判断手段において、高周波成分と判断された場合に、誤り訂正できなかった復号データに対応する変換係数のみを0で置き換える修整手段とを備えることにより、上記課題を解決する。

【0019】本発明の第6の発明によれば、ブロック毎の画素値を直交変換して求められる変換係数を量子化した量子化インデックスを含むデータを符号化した符号化データを入力して誤り訂正復号を行い、復号データを出力する誤り訂正復号手段と、前記誤り訂正復号手段において、誤り検出はできたが、誤り訂正できなかった復号データが、水平方向に隣接するブロックと相関の高い周波数成分、或いは、垂直方向に隣接するブロックと相関の高い周波数成分であるか否かを判断する判断手段と、前記判断手段において、水平方向に隣接するブロックと相関の高い周波数成分と判断された場合には、誤り訂正できなかった復号データに対応する変換係数のみを、水平方向に隣接するブロックの同一周波数成分の変換係数で置き換え、垂直方向に隣接するブロックと相関の高い周波数成分と判断された場合には、誤り訂正できなかった復号データに対応する変換係数のみを、垂直方向に隣接するブロックの同一周波数成分の変換係数で置き換える修整手段とを備えることにより、上記課題を解決する。

【0020】

【発明の実施の形態】以下、図面を参照して本発明の動画像変換符号化誤り修整装置の実施例を詳細に説明する。

【0021】図1は、本発明の動画像変換符号化誤り修整装置（以下、誤り修整装置と称する）の一実施例の構成を示すブロック図である。

【0022】図1の誤り修整装置は、入力デジタル動画像データに離散コサイン変換を行い、得られた変換係数を量子化し、量子化インデックスを量子化ビット数を表す付加情報と共に誤り訂正符号化を行って記録媒体に記録し、記録データを読み出し、読み出したデータを誤り復号して、誤りが検出されたデータに本実施例の誤り修整方法を適用して復号を行うように構成されている。

【0023】図1の誤り修整装置は、符号化器10、記録媒体11、及び記録媒体11を介して符号化器10に接続された復号器12によって構成されている。

【0024】符号化器10は、入力デジタル動画像データを8画素×8画素のブロック毎に並べ替えるブロック化回路13、ブロック化回路13に接続されておりブロック化回路13で並べ替えられた動画像データを入力して離散コサイン変換して8×8の変換係数を一定の規則にしたがって出力する離散余弦変換(DCT)回路14、DCT回路14に接続されており入力データを付加情報に基づいて適応的に量子化して量子化インデックスを出力する量子化器15、量子化器15に接続されており付加情報及び量子化インデックスを入力して誤り訂正符号化を行なう誤り訂正符号化器16によって構成されている。

【0025】誤り訂正符号化器16で誤り訂正符号化された結果は、磁気テープや磁気ディスクなどの記録媒体11に記録される。

【0026】また、図1に示すように復号器12は、記録媒体11に接続されており記録媒体11から読み出したデータを入力してワード単位で誤り訂正復号を行なう誤り訂正復号器17、誤り訂正復号器17に接続されており入力データを付加情報と量子化インデックスに分離し、付加情報と量子化インデックスから得られる逆量子化値(DCT係数値)を出力線cに出力し、付加情報に誤りが生じている場合にはエラーフラッグdをONにし、誤ったDCT係数値を出力する場合には誤りフラッグeをONにする逆量子化器18、逆量子化器18に接続されており誤ったDCT係数値が入力された場合に、第1周波数成分のデータである領域Aに属している場合には誤りフラッグgをONにし、第2周波数成分のデータである領域Bに属している場合は誤っている係数値を0で置き換え、第3周波数成分のデータである領域Cに属している場合は誤っている係数値を水平方向の隣接ブロックの同一周波数成分の係数値で置き換え、第4周波数成分のデータである領域Dに属している場合は誤っている係数値を垂直方向の隣接ブロックの同一周波数成分の係数値で置き換え、付加情報に誤りが生じている場合は、全ての係数値

が誤っていることになり、誤り修整では良好な再生画像は得られないため、領域Aの係数に誤りが生じている場合と同様の処理をする第1修整手段である第1修整回路19、第1修整回路19に接続されており第1修整回路19からの入力値fにしたがってブロック毎に逆DCTを行い、ブロック毎の再生画素値hを出力する逆DCT回路20、第1修整回路19に接続されており第1修整回路19からの出力データfが逆DCT回路20で遅延する時間分だけ入力信号を遅延させる遅延回路21、逆DCT回路20及び遅延回路21に接続されており誤りフラッグg'がOFFの場合には、hから入力された再生画素値をメモリに書き込むと共に、入力された値をそのまま再生画素値としてブロック分解回路23(後述)に出力し、誤りフラッグがONの場合には、入力hからの再生画素値を用いずにメモリから前画面の同位置の再生画素値をブロック分解回路23(後述)に出力する第2修整手段である第2修整回路22、第2修整回路22に接続されており第2修整回路22からの入力値を、符号化器10のブロック化回路13に入力されたデジタル動画データと同様のフォーマットで出力するブロック分解回路23によって構成されている。上記第1修整回路19の説明で用いた領域A~Dの一例を図10に示す。

【0027】次に、上記各構成部分を詳述する。

【0028】符号化器10を構成しているブロック化回路13は、入力されたデジタル動画データを8画素×8画素のブロック毎に並べ替える。

【0029】DCT回路14は、ブロック化回路13に接続されておりブロック化回路13で並べ替えられた動画データをを入力して離散コサイン変換して8×8の変換係数を一定の規則にしたがって出力する。

【0030】量子化器15は、DCT回路14に接続されており入力データを付加情報に基づいて適応的に量子化して量子化インデックスを出力する。

【0031】誤り訂正符号化器16は、量子化器15に接続されており付加情報及び量子化インデックスを入力して誤り訂正符号化を行なう。

【0032】誤り訂正復号器17は、記録媒体11に接続されており記録媒体11から読み出したデータを入力してワード単位で誤り訂正復号を行なう。

【0033】逆量子化器18は、誤り訂正復号器17に接続されており入力データを付加情報と量子化インデックスに分離し、付加情報と量子化インデックスから得られる逆量子化値(DCT係数値)を出力線cに出力し、付加情報に誤りが生じている場合にはエラーフラッグdをONにし、誤ったDCT係数値を出力する場合には誤りフラッグeをONにする。

【0034】第1修整回路19は、逆量子化器18に接続されており誤ったDCT係数値が入力された場合に、入力係数値が領域Aに属している場合には誤りフラッグgをONにし、領域Bに属している場合は誤っている係数値

を0で置き換え、領域Cに属している場合は誤っている係数値を水平方向の隣接ブロックの同一周波数成分の係数値で置き換え、領域Dに属している場合は誤っている係数値を垂直方向の隣接ブロックの同一周波数成分の係数値で置き換え、付加情報に誤りが生じている場合は、全ての係数値が誤っていることになり、誤り修整では良好な再生画像は得られないため、領域Aの係数に誤りが生じている場合と同様の処理をする。なお、第1修整回路19では、領域Cと領域Dの係数に誤りが生じた場合、誤った係数値を過去の隣接ブロックの係数値との置き換えるため、領域Cと領域Dの周波数成分の係数値をメモリに格納している。

【0035】逆DCT回路20は、第1修整回路19に接続されており第1修整回路19からの入力値fにしたがってブロック毎に逆DCTを行い、ブロック毎の再生画素値hを出力する。

【0036】遅延回路21は、第1修整回路19に接続されており第1修整回路19からの出力データfが逆DCT回路20で遅延する時間分だけ入力信号を遅延させる。

【0037】第2修整回路22には、誤りフラッグg'と、逆DCT回路20からブロック毎の再生画素値hが入力されており、1画面分の再生画素値を記録できるメモリを備えている。誤りフラッグg'は第1修整回路19から出力された誤りフラッグgを遅延回路21で遅延させた信号である。第2修整回路22では、誤りフラッグg'がOFFの場合には、hから入力された再生画素値をメモリに書き込むと共に、入力された値をそのまま再生画素値としてブロック分解回路23に出力する。誤りフラッグがONの場合には、入力hからの再生画素値は用いず、メモリから前画面の同位置の再生画素値をブロック分解回路23に出力する。

【0038】ブロック分解回路23は、第2修整回路22に接続されており第2修整回路22からの入力値を符号化器10のブロック化回路13に入力されたデジタル動画データと同様のフォーマットで出力する。

【0039】図2は、図1の逆量子化器18による1ブロック分のデータの逆量子化の手順を説明するためのフローチャートである。

【0040】逆量子化器18の入力線aからは8ビットのデータが入力され、入力線bは入力線aから入力されたデータに誤りが生じているときにのみONとなる。

【0041】出力線cからは入力データから分離された付加情報や量子化インデックスが出力される。

【0042】図3は、逆量子化器18に入力されるデータをビット毎に示す。

【0043】図3で、kはクラス分け情報のビット数、pはビットパターン情報のビット数、b(m)はm番目の係数の量子化ビット数を示している。なお、m番目の係数の量子化ビット数b(m)の値はクラス分け情報とビットパターン情報とから決定できる。

【0044】図2では入力線aから入力される8ビットのデータをMSB（最上位ビット）からD（0）～D（7）とし、入力値を付加情報と係数の量子化インデックスとに分離し、付加情報と係数の逆量子化値を出力線cに出力する流れを示している。図2でも図3と同様にkはクラス分け情報のビット数、pはビットパターン情報のビット数、b（m）はm番目の係数の量子化ビット数を示し、mは64個の量子化値の何番目かを示している。

【0045】まず、出力エラーフラッグd、eをOFF *10

$$K = 2^{k-1} \times K(0) + 2^{k-2} \times K(1) + \dots + 2^0 \times K(k-1)$$

$$P = 2^{p-1} \times P(0) + 2^{p-2} \times P(1) + \dots + 2^0 \times P(p-1)$$

である。このとき、必要なビット数が8ビットより多い場合は（ステップS3、及び後述するステップS8）、上記ステップS2及び後述するステップS7で逐次、入力線aから8ビット毎に情報を得て、エラーフラッグdのON、OFFを設定する。続いて、入力データを1ビットずつKに格納し（ステップS4）、j < kがNOとなったときにkビットのクラス分け情報の入力が終了して（ステップS5）、Kをcに出力する（ステップS6）。なお、ステップS7は、上記ステップS2と同様の動作を行う。

【0049】入力データを1ビットずつPに格納し（ステップS9）、j < pがNOとなったときにpビットのビットパターン情報の入力が終了し（ステップS10）、Pをcに出力する（ステップS11）。この時点で付加情報が得られているので、どの係数が何ビットで量子化されているかがわかる。

【0050】次に64個の量子化インデックスを順次読み出しQに格納する。ステップS12～S18で1つの係数の処理をしている。b（m）=0がYES、すなわち0ビット量子化を行っている場合か（ステップS15）、またはj < b（m）がNOの場合に、m番目の量子化値の入力が終了する（ステップS17）。そしてQを逆量子化し、その値を出力線cに出力し、mをインクリメントすると共にQをクリアする（ステップS18）。そして、m < 64がNOとなったときに1ブロック分の64個の逆量子化値の出力が終了する（ステップS19）。ここでも入力データは逐次ステップS13から入力される。ステップS13の詳細な動作は、図5のフローチャートに示している。

【0051】図2のステップS2とステップS7の動作を図4のフローチャートで示す。

【0052】入力線aからは8ビットのデータが入力され、エラーフラッグbは入力線aから入力されたデータに誤りが生じている時にONとなる。

【0053】まず、aから誤り訂正復号された8ビットのデータを入力し、MSBからD（0）～D（7）とする（ステップS21）。入力データに誤りが生じている

*F、m=0、j=0とし（ステップS1）、入力線aから8ビットの情報を入力し、出力エラーフラッグdを設定する（ステップS2）。

【0046】ステップS2の詳細な動作を図4のフローチャートに示す。

【0047】次に、上記ステップS2で得られた入力データからクラス分け情報をKに、ビットパターン情報をPに inputsする。ここで、

【0048】

【数1】

場合には入力エラーフラッグbがONとなっているが（ステップS22）、その場合には出力エラーフラッグdをONとし（ステップS23）、そうでない場合には出力エラーフラッグdをOFFとする（ステップS24）。そして、Dのビット番号を示すiを0とする（ステップS25）。

【0054】図2のステップS13の動作を図5のフローチャートで示す。

【0055】ここでも、図4と同様に入力線aからは8ビット毎にデータが入力され、エラーフラッグbは入力線aから入力されたデータに誤りが生じている時にONとなる。

【0056】まず、入力線aから誤り訂正復号された8ビットのデータを入力し、MSBからD（0）～D（7）とする（ステップS31）。入力データに誤りが生じている場合には入力エラーフラッグbがONとなっているが（ステップS32）、その場合には出力エラーフラッグeをONとし（ステップS33）、そうでない場合には出力エラーフラッグeをOFFとする（ステップS34）。そして、Dのビット番号を示すiを0とする。

【0057】図6は、第1修整回路19で1ブロック分のDCT係数値を修整する動作のフローチャートを示す。

【0058】まず、出力エラーフラッグgをOFFにし、出力線cからクラス分け情報を入力する（ステップS41）。次に、入力エラーフラッグdの状態を判定し、エラーフラッグdがONの場合には、入力されたクラス分け情報が誤りであることを示している（ステップS42）、エラーフラッグgをONにする（ステップS43）。

【0059】次に、出力線cからビットパターン情報を入力し（ステップS44）、エラーフラッグdの状態を判定する（ステップS45）。エラーフラッグdがONの場合には入力ビットパターン情報が誤りであることを示している（ステップS46）。上述のステップS41～S46で、付加情報に誤りが生じている場合にエラーフラッグgがONと

なっている。

【0060】続いて、何番目のDCT係数を扱っているかを示す m を0とする(ステップS47)。そして m が64になるまで各係数について、後述するステップS48からステップS60までの処理を繰り返す。

【0061】まず、出力線 c から m 番目のDCT係数値を入力し(ステップS48)、入力エラーフラグ e の状態を判定する(ステップS49)。エラーフラグ e がOFFの場合は、入力された係数値は正しい値ということを示している。エラーフラグ e がOFFの場合、係数値がC領域かD領域に属する場合は、今後誤りが生じた場合に置き換えに用いることができるように、入力係数値をメモリに格納しておく。領域A、領域Bの係数の場合は、置き換えに用いないので、係数値を格納しておく必要はない。上記ステップS49でエラーフラグ e がONの場合は入力された係数値が誤っていることを示しているので、入力係数値がどの領域に属するかを判定し、領域Aに属す場合はエラーフラグ g をONにし(ステップS53)、領域Bに属す場合は係数値を0とし(ステップS55)、領域Cに属す場合は水平方向の隣接ブロックの同一周波数成分の係数値で置き換え(ステップS57)、領域Dに属す場合は垂直方向の隣接ブロックの同一周波数成分の係数値で置き換える(ステップS58)。

【0062】次に、正しい係数値、または修整後の係数値を出力線 f に出力し、 m をインクリメントする(ステップS59)。 m が64となる、すなわち1ブロックの全ての係数値について修整が終了するとループを抜ける(ステップS60)。

【0063】図7は、第2修整回路22で1ブロック分の再生画素値を修正する動作のフローチャートを示す。

【0064】出力線 h からはブロックの各画素の再生画素値が入力され、 g' はブロックの画素値を前画面の再生画素値で置き換えるかどうかを示すフラグである。

【0065】まず、何番目の画素を扱っているかを示す n を0とし(ステップS61)、次にフラグ g' の状態を調べ(ステップS62)、フラグ g' がOFFの場合、即ち、A領域に含まれる係数値または付加情報に誤りが生じていない場合は、入力される再生画素値をそのまま出力するため後述するステップS63～ステップS65を繰り返す。

【0066】入力線 h から n 番目のデータを入力し(ステップS63)、入力データをメモリに格納すると共にブロック分解回路23に出力して n をインクリメントする(ステップS64)。そして、 n が64になる、即ち、1ブロックのデータを処理し終わるとループを抜ける(ステップS65)。

【0067】上記ステップS62でフラグ g' がONの場合、即ち、A領域に含まれる係数値または付加情報に誤りがあった場合は、メモリに格納されている前画面

の同位置の再生画素値を読み出すため、後述するステップS66～ステップS68を繰り返す。

【0068】まず、メモリから前画面の n 番目のデータを読み出し(ステップS66)、次に読み出したデータをブロック分解回路23に出力すると共に、 n をインクリメントする(ステップS67)。 n が64になる、即ち、1ブロックのデータを処理し終わるとループを抜ける(ステップS68)。

【0069】図8は、第2修整回路22の他の実施例として、1ブロック分の再生画素値を修正する際に、動き検出を行ってより良好な再生画像を得る第2修整回路22の動作のフローチャートを示す。

【0070】図7と同様に、出力線 h からはブロックの各画素の再生画素値が入力され、 g' はブロックの画素値を前画面の再生画素値で置き換えるかどうかを示すフラグである。

【0071】図8のステップS71～S75は図7のステップS61～S65と同一である。フラグ g' がONの場合、即ち、領域Aに含まれる係数値または付加情報に誤りがあった場合には(ステップS72)、過去の再生ブロックから動き検出を行い、動きベクトルを(i, j)とする(ステップS79)。次にメモリに格納されている前画面の再生画素値を読み出すため、ステップS76～ステップS78を繰り返す。

【0072】まず、メモリから前画面の n 番目のデータから(i, j)だけずらした位置のデータを読みだし(ステップS76)、読み出したデータをブロック分解回路23に出力すると共に、 n をインクリメントする(ステップS77)。上記ステップS76で n 番目のデータがメモリ上にアドレス(k, l)に格納されているとすると、アドレス($k+i, l+j$)のデータを読み出すことになる。 n が64になる、即ち1ブロックのデータを処理し終わるとループを抜ける(ステップS78)。

【0073】図9は、本発明の誤り修整方法を用いた装置の他の実施例の一構成例のブロック図である。

【0074】図9の装置と上述した図1の装置との違いは用いられている誤り修整方法にある。

【0075】図1の装置では、誤りの生じた係数によって、第1修整回路19で誤りの生じたDCT係数値を修正する場合と、第2修整回路22で誤りの生じたブロックの画素値を前画面の画素値を用いて置き換えを行う場合とに分けていた。

【0076】図9の装置では、全て誤りの生じたDCT係数値を修整する。

【0077】なお、図9のブロック化回路31～逆量子化器37は図1のブロック化回路13～逆量子化器18及び記録媒体11と、図9の逆DCT回路39は図1の逆DCT回路20と、図9のブロック分解回路40は図1のブロック分解回路23と同一である。図9においてブロック化回路31～誤り訂正符号化器34が符号化器41、誤り訂正復号器36～

ブロック分解回路40が復号器42をそれぞれ構成する。

【0078】以下、図1との相違点である修整手段である修整回路38の動作を詳細に説明する。

【0079】図1の第1修整回路19と同様に、入力される64個の係数を図10の領域A～Dに分け、それぞれの領域A～Dに適した修整方法を用いる。

【0080】入力されるデータも図1の第1修整回路19と同一で、 c' からは付加情報とDCT係数値が入力され、 d' は c' から入力された付加情報に誤りが生じている場合にONとなるエラーフラッグで、 e' は c' から入力されたDCT係数値に誤りがある場合にONとなるエラーフラッグである。

【0081】なお、修整回路38には修正に用いるため、領域A、C、DのDCT係数値がメモリに格納されている。

【0082】まず、第5周波数成分のデータである領域Aの係数値に誤りが生じた場合、誤りの生じた係数値を前画面の同位置のブロックの同一周波数成分の係数値で置き換える。第6周波数成分のデータである領域Bの係数値、第7周波数成分のデータである領域Cの係数値、及び第8周波数成分のデータである領域Dの係数値にそれぞれ誤りが生じた場合は、図1の第1修整回路19と同様の動作を行う。即ち、領域Bの係数値に誤りが生じた場合、誤りの生じた係数値を0とし、領域Cの係数値に誤りが生じた場合、誤りの生じた係数値を水平方向の隣接ブロックの同一周波数成分の係数値と置き換え、領域Dの係数値に誤りが生じた場合、誤りの生じた係数値を垂直方向の隣接ブロックの同一周波数成分の係数値と置き換える。付加情報に誤りが生じた場合は、全ての係数値に誤りが生じたものとして、全ての係数値に上記の処理を行う。

【0083】図11は、修整回路38で1ブロック分のデータを処理する動作のフローチャートである。 c' からは付加情報と64個のDCT係数値が入力され、 d' は入力される付加情報に誤りが生じているかどうかを示すエラーフラッグ、 e' はDCT係数値に誤りが生じているかどうかを示すエラーフラッグである。修正されたDCT係数値は f' から出力される。

【0084】まず、 c' からクラス分け情報を入力し(ステップS81)、入力エラーフラッグ d' の状態を判定する(ステップS82)。 d' がONの場合、即ち入力されたクラス分け情報が誤りであるには、ステップS83に分岐し、 c' からビットパターン情報を入力する。 d' がOFFの場合は、 c' からビットパターン情報を入力し(ステップS84)、 d' の状態を判定する(ステップS85)。 d' がONの場合は入力ビットパターン情報が誤りであることを示しているため、ステップS94に分岐する。

【0085】以下、付加情報に誤りがない場合はステップS86～S92を実行し、誤りがある場合はステップ

S94～S98を実行することになる。

【0086】付加情報に誤りのない場合は、何番目のDCT係数を扱っているかを示す m を0とする(ステップS86)。そして m が64になるまで各係数についてステップS87～S92の処理を繰り返す。

【0087】まず、 c' から m 番目DCT係数値を入力し(ステップS87)、入力エラーフラッグ e' の状態を判定する(ステップS88)。 e' がONの場合は入力された係数値が誤っていることを示しているため、誤った係数値の修整を行う(ステップS93)。

【0088】図12のフローチャートは、上記ステップS93の詳細な動作を示す。

【0089】 e' がOFFの場合、即ち、入力されたDCT係数値に誤りがない場合は、係数値の領域を判定し(ステップS89)、領域A、C、Dの場合には係数値をメモリに格納する(ステップS90)。そうでない場合にはステップS91にジャンプして、入力された正しい係数値、または修整された係数値が f' から出力される(ステップS91)。そして、 $m < 64$ がNOとなるまでステップS87～S91を繰り返し(ステップS92)、上記ステップS92の判定結果がNOとなると、ループを抜け終了する。

【0090】付加情報に誤りがあった場合は、何番目のDCT係数を扱っているかを示す m を0とする(ステップS94)。そして m が64になるまで各係数についてステップS95～S98の処理を繰り返す。

【0091】まず、 c' から m 番目のDCT係数値を入力し(ステップS95)、係数値の修整を行う(ステップS96)。ステップS94～S98に進むのは、付加情報に誤り生じた場合であり、付加情報に誤りが生じた場合は、全てのDCT係数値に誤りが生じる。そのためステップS95で入力したDCT係数値全てをステップS96で修整する。なお、ステップS96はステップS93と同一の動作を行う。

【0092】修整された係数値が f' から出力される(ステップS97)。そして、 $m < 64$ がNOとなるまでステップS95～S97を繰り返し(ステップS98)、ステップS98の判定結果がNOとなると、ループを抜け終了する。

【0093】図12のフローチャートは、図11のステップS93、ステップS96の詳細な動作を示す。ここでは誤りの生じているDCT係数値が入力され、その領域毎に適応的な処理を行う。

【0094】まず、入力係数値が領域Aに属するかどうかを判定し(ステップS101)、領域Aに属す場合は誤った係数値を前画面の同位置のDCT係数値と置き換える(ステップS102)。領域Aに属さない場合は、入力係数値が領域Bに属するかどうかを判定する(ステップS103)。領域Bに属す場合は誤った係数値を0とする(ステップS104)。領域Bに属さない場合

は、入力係数値が領域Cに属するかどうかを判定する(ステップS105)。領域Cに属す場合は水平方向の隣接ブロックの同一周波数成分の係数値で置き換える(ステップS106)。領域Cに属さない場合は、入力係数値は領域Dに属するので垂直方向の隣接ブロックの同一周波数成分の係数値で置き換える(ステップS107)。

【0095】なお、本発明は、1フレーム、または1フィールドをブロックに分割し、そのブロック毎に変換を行い、高能率に符号化された符号列に対して行うものであり、本実施例のような離散コサイン変換を行った後の変換係数に限らず、KL変換、ウェーブレット変換等どのような変換が行われてもよい。

【0096】また、量子化方法も、上記の実施例は一例として示したものであり、どのような方法で量子化が行われてもよい。

【0097】本発明では、読み出したデータに誤りが生じた場合、どの変換係数に誤りが生じたかで修整方法を適応的に選択する。変換符号化の一例として、8画素×8画素のブロックにDCT(離散コサイン変換)を用いた場合の手段を示す。

【0098】8画素×8画素のブロックにDCTを行なった場合、図13にあるように、64個の変換係数が得られ、それぞれがブロックの画素値の空間的な周波数成分を表している。図13のように水平方向にu軸、垂直方向にv軸をとると、 $u=0$ 、 $v=0$ のとき直流成分を表す。 $u=1$ のとき水平方向の最低周波数成分を表し、順に高周波数成分を表し、 $u=7$ のとき最高周波数成分を表す。 $v=1$ のときは垂直方向の最低周波数成分を表し、順に高周波数成分を表し、 $v=7$ のとき最高周波数成分を表す。まず、直流成分や低周波数成分に誤りが生じた場合を考える。

【0099】8×8のマトリック上で直流成分や低周波数成分の位置の一例を示すと図14の斜線部分のようになる。

【0100】直流成分や低周波数成分は、統計的にみるとその値が大きく、隣接ブロック間の同一周波数成分で大きな相関があり、通常、その相関を利用して予測を行うと「TV学会技報」, Vol12, No. 10, pp. 19~24 Feb. 1988に示されているような問題が生じてしまうが、本発明ではこの問題も解決されている。

【0101】ところで、動画像データの場合時間的な相関も非常に大きい。そこで予測値として、前画面の同位置の再生画像を用いれば、1ブロックとしては良好な再生画像が得られる。すなわち、直流成分や低周波数成分に誤りが生じた場合は、復号後、再生画像を前画面の同位置の再生画素値と置き換えると前記の問題点は解決できる。隣接ブロックを用いて動き検出を行い、前画面の再生画像を動きベクトル分だけずらしたもので置き換えても良い。また、1ブロック全体を置き換えるのではな

く、逆DCT前に誤りが生じた係数だけを前画面の同位置のブロックのDCT係数値で置き換えても良好な再生画像が得られる。

【0102】次に、高周波数成分に誤りが生じた場合を考える。8×8のマトリック上で高周波数成分の位置の一例を示すと図15の斜線部分のようになる。

【0103】高周波数成分は、空間的にも時間的にも相関が低く、相関を用いて予測を行うのは困難である。しかしながら、統計的にみるとその値は低周波数成分に比較して小さく、誤りが生じた場合でも、画像の細かな変化が変わるだけで、視覚的にも大きな影響は与えない。例えば高周波数成分の値を0としたときは、画像の細かな変化がなくなるため、画像が若干なまる感じになるが、概要には変化がない。したがって、高周波数成分に誤りが生じた場合には、その係数値を0で置き換えるとよい。また、画像データには連続するエッジをもったものが多いが、例えば図16のように水平方向に連続するエッジをもった画像データに対して直交変換を行った場合、図中のブロック1、ブロック2のように水平方向に隣接するブロックの変換係数のうち、垂直方向の周波数成分には非常に大きな相関がある。

【0104】8×8のマトリック上で水平方向に隣接するブロックの変換係数と大きな相関がある係数の位置の一例を示すと図17の斜線部分になる。

【0105】また、図18のような垂直方向に連続するエッジをもった画像データに対しては、図中のブロック3、ブロック4のように垂直方向に隣接するブロックの変換係数のうち、水平方向の周波数成分に非常に大きな相関がある。

【0106】8×8のマトリック上で垂直方向に隣接するブロックの変換係数と大きな相関がある係数の位置の一例を示すと図19の斜線部分になる。

【0107】このようにブロック間で非常に大きな相関を持った係数については、隣接ブロックの値で置き換えることが有効である。これらのことから、上述したように、直流成分や低周波数成分で構成される領域を領域A、高周波数成分で構成される領域を領域B、水平方向の隣接ブロックの係数値と相関の大きい係数で構成される領域を領域C、垂直方向の隣接ブロックの係数値と相関の大きい係数で構成される領域を領域Dとすると、領域Aに属する係数値に誤りが生じた場合は、周波数成分での修整は行わず、逆DCT後に再生画面上で、誤りの生じているブロックの再生画素値を、誤りが生じたブロックと同位置の画素値で置き換え、あるいは動き検出を行い、動きベクトル分だけずらした位置の画素値で置き換え、領域Bに属する係数値に誤りが生じた場合は、誤りの生じた係数値を0とし、領域Cに属する係数値に誤りが生じた場合は、誤りの生じた係数値を水平方向の隣接ブロックの同一周波数成分の値と置き換えを行い、領域Dに属する係数値に誤りが生じた場合は、誤りの生じ

た係数値を垂直方向の隣接ブロックの同一周波数成分の値と置き換えを行う。または、領域Aに属する係数値に誤りが生じた場合に、誤った係数値を前画面の同位置のブロックのDCT係数値で置き換える。

【0108】 8×8 マトリックス上でそれぞれの領域の一例を図示すると、図10のようになる。このように周波数成分に応じた修整方法を用いることにより、誤りが生じた場合でも良好な再生画像が得られる。

【0109】なお、上記の各領域は一例として示したので、入力データの形式、サブサンプリングの有無、インターレースされているかどうかなどで最適な領域は異なる。また、上記は変換としてDCTを用いた場合の一例であり、異なる変換を用いると、最適な領域は異なる。

【0110】本発明によると、読み出したデータに誤りが検出された時には、誤りが生じた係数によって、以下の誤り修正を行う。視覚的に大きな影響を与えるデータに誤りが生じた場合には、誤りが生じたブロックの再生画素値を前画面の再生画素値で置き換える、または誤った係数を前画面の同位置の係数で置き換える。

【0111】視覚的に大きな影響を与えないデータに誤りが生じた場合には、その値を0とする。

【0112】比較的大きな影響を与え、なおかつ周辺ブロックの係数に強い相関をもつ係数に誤りが生じた場合は、隣接ブロックの係数値と置き換える。そのため、データに誤りが生じた場合にも、小さい誤差で、かつ画像がなまることなく再生でき、良好な再生画像が得られる。

【0113】

【発明の効果】本発明によれば、誤り訂正復号を行う際に、誤り検出はできたが、誤り修整できなかつた符号化データのデータの種別を判定して、該データの種別に基づいて最適な誤りの修整を行うため、時間的、空間的に最適な誤り修整方法が選択され、その結果、良好な再生画像が得られる。

【図面の簡単な説明】

【図1】本発明における復号装置の一実施例の構成を示すブロック図である。

【図2】図1の逆量子化器の動作を示すフローチャートである。

【図3】図1の逆量子化器に入力されるデータの1例である。

【図4】図2のステップS2, S7の動作を示すフローチャートである。

【図5】図2のステップS13の動作を示すフローチャートである。

【図6】図1の第1修整回路の動作を示すフローチャートである。

【図7】図1の第2修整回路の動作を示すフローチャートである。

【図8】動き検出を用いた第2修整回路の動作を示すフローチャートである。

【図9】本発明における復号装置の他の実施例の構成を示すブロック図である。

【図10】DCT係数の相関関係の説明図である。

【図11】図9の修整回路の動作を示すフローチャートである。

【図12】図11のステップS93, S96の動作を示すフローチャートである。

【図13】 8×8 のDCT係数の説明図である。

【図14】DCT係数の相関関係の説明図である。

【図15】DCT係数の相関関係の説明図である。

【図16】水平方向に連続したエッジをもつ画像の説明図である。

【図17】DCT係数の相関関係の説明図である。

【図18】垂直方向に連続したエッジをもつ画像の説明図である。

【図19】DCT係数の相関関係の説明図である。

【図20】従来の技術の説明図である。

【符号の説明】

10 符号化器

11 記録媒体

12 復号器

13 ブロック化回路

14 DCT回路

15 量子化器

16 誤り訂正符号化器

17 誤り訂正復号器

18 逆量子化器

19 第1修整回路

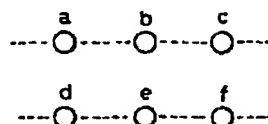
20 逆DCT回路

40 21 遅延回路

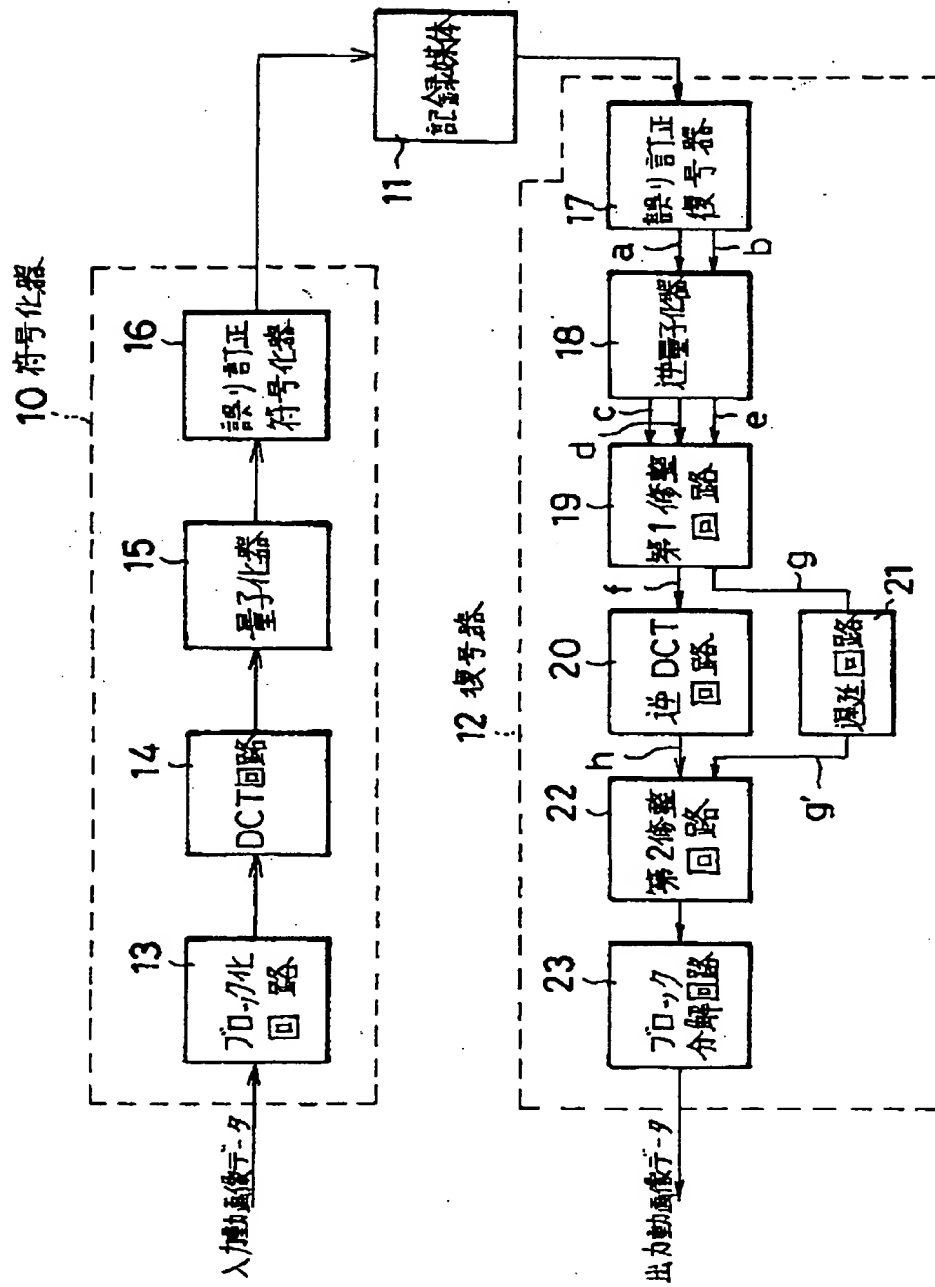
22 第2修整回路

23 ブロック分解回路

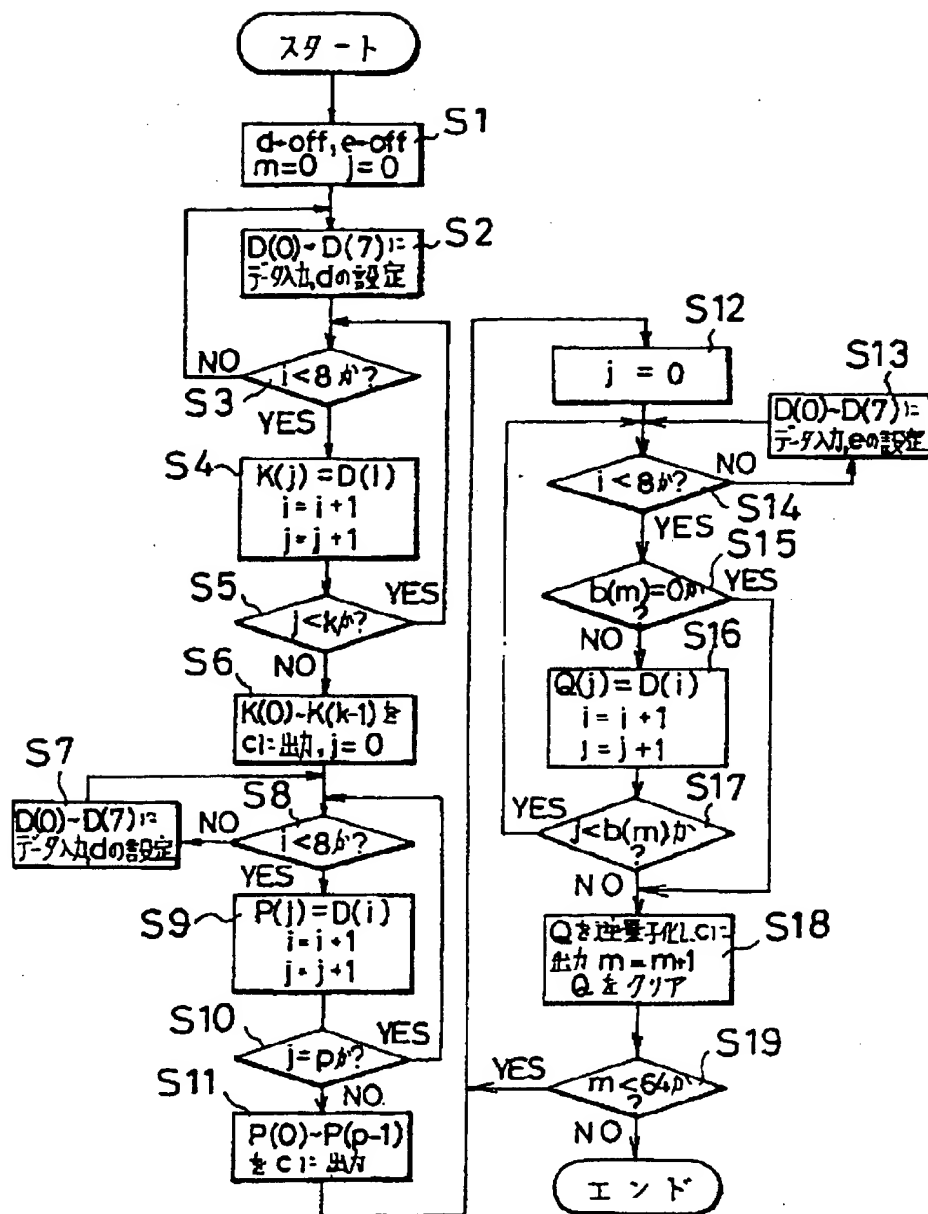
【図20】



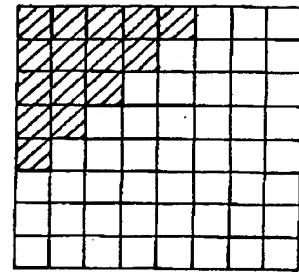
【図1】



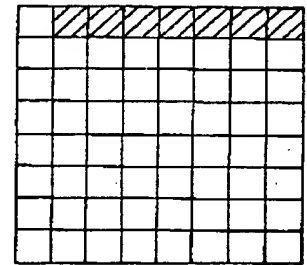
【図2】



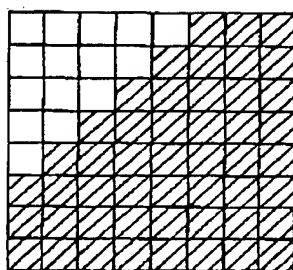
【図14】



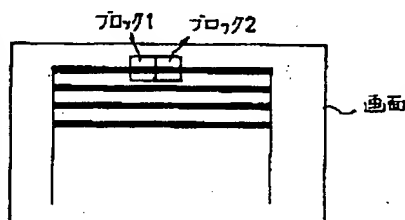
【図19】



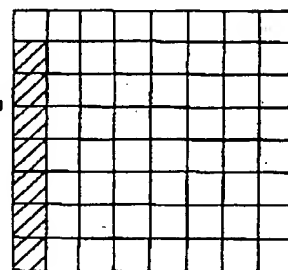
【図15】



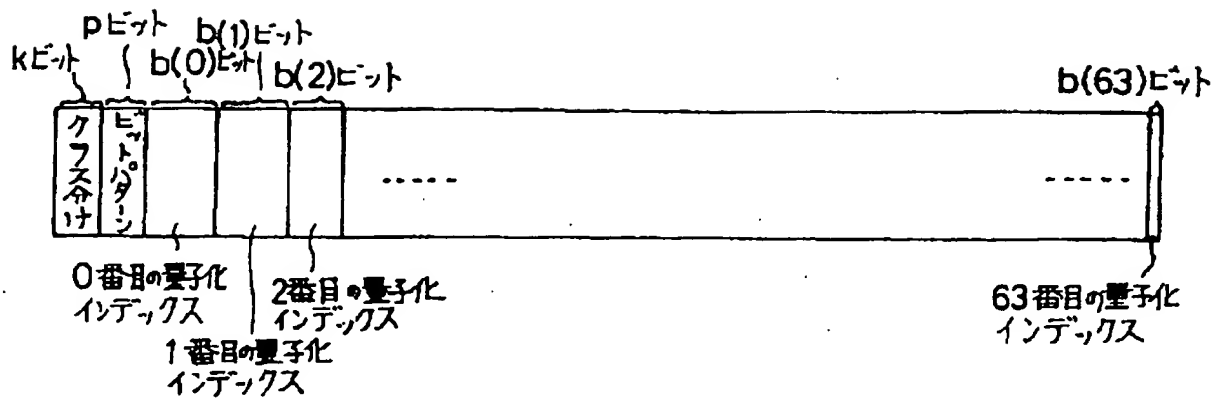
【図16】



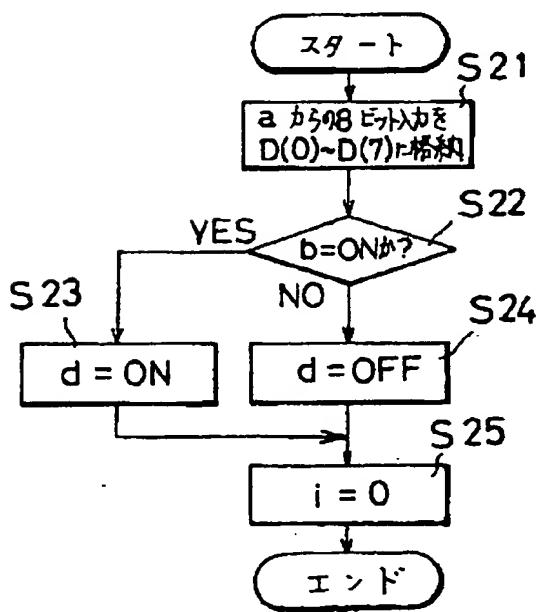
【図17】



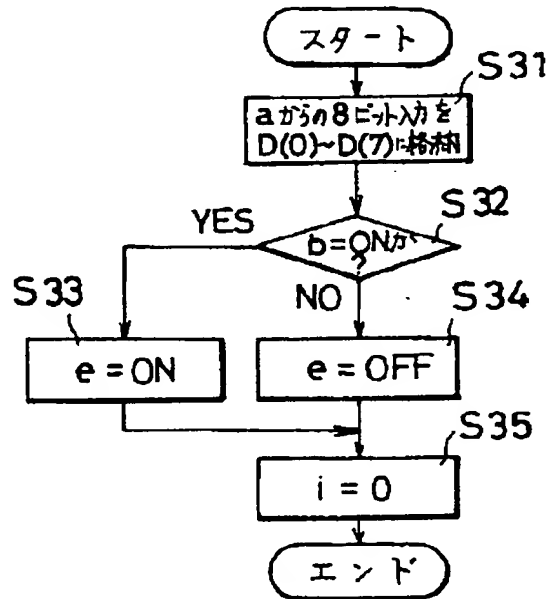
【図3】



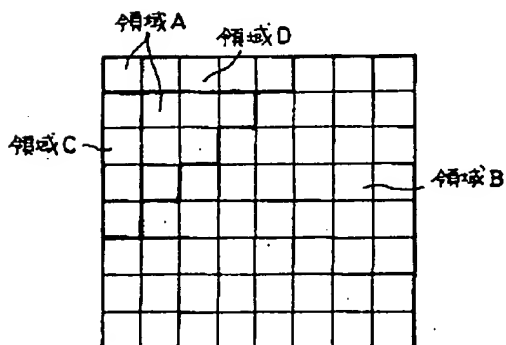
【図4】



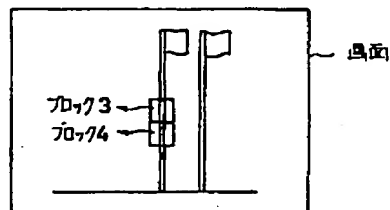
【図5】



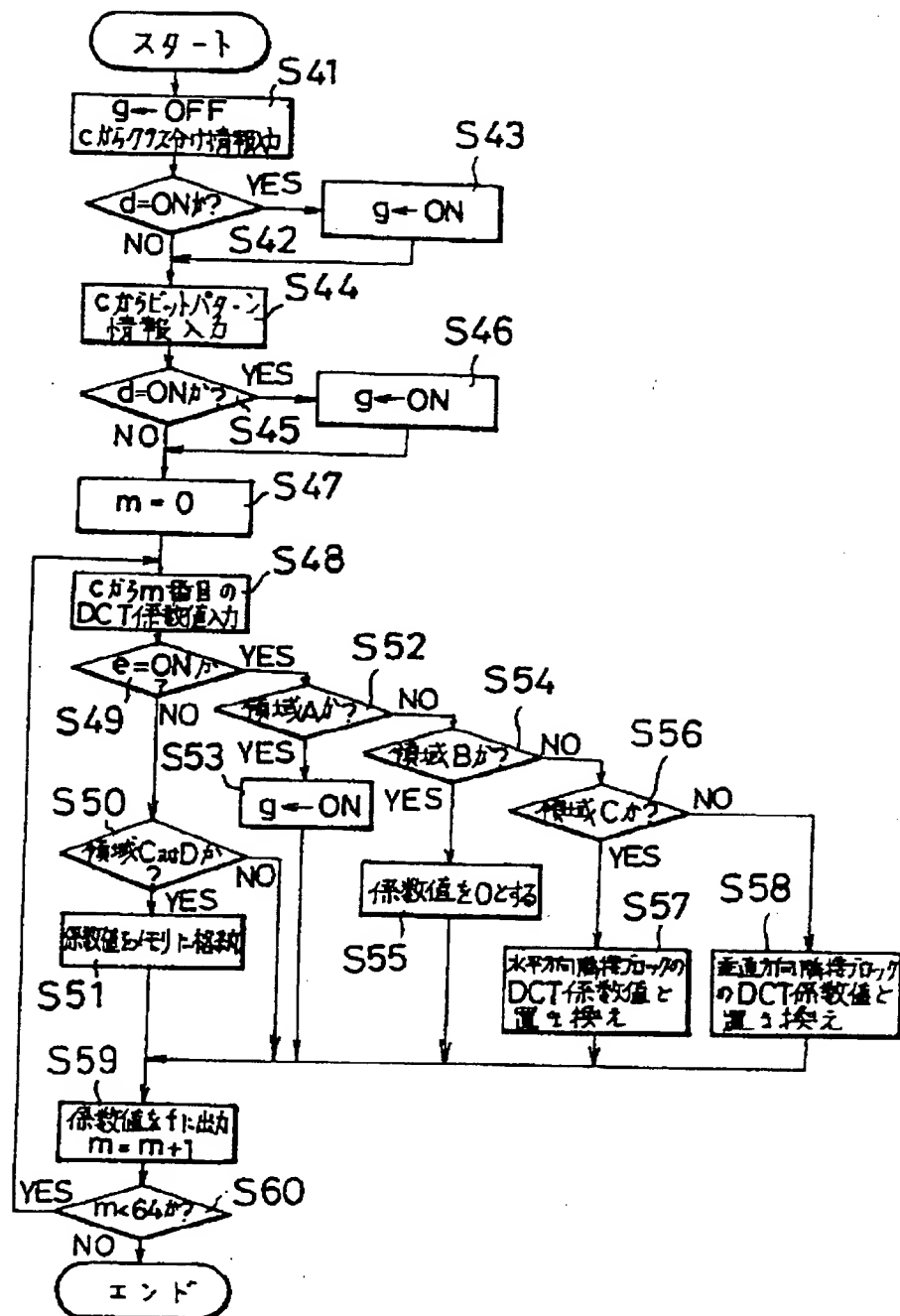
【図10】



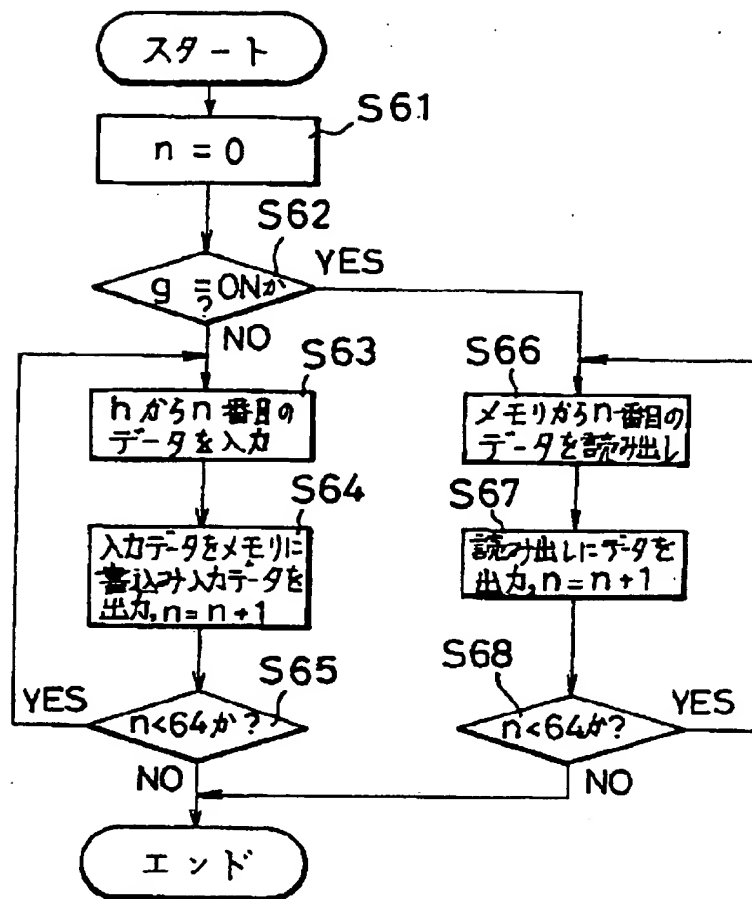
【図18】



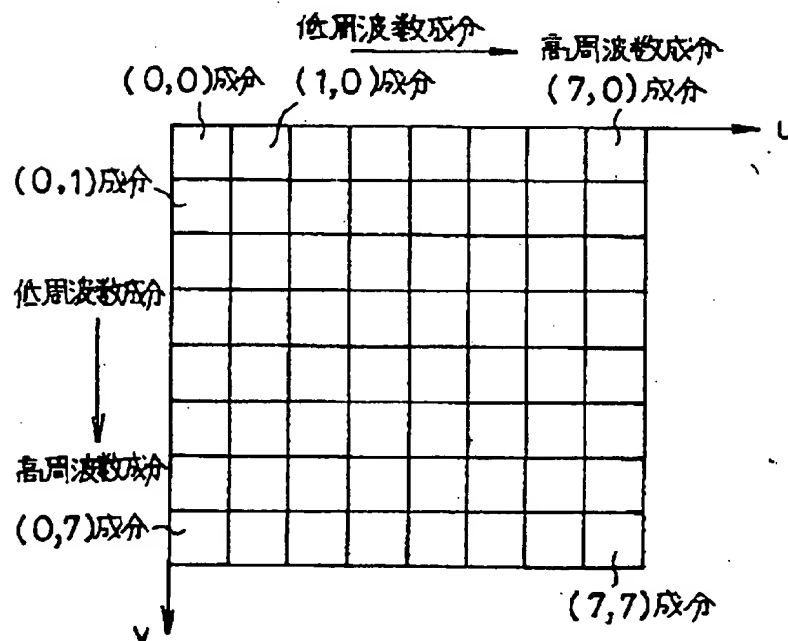
【図6】



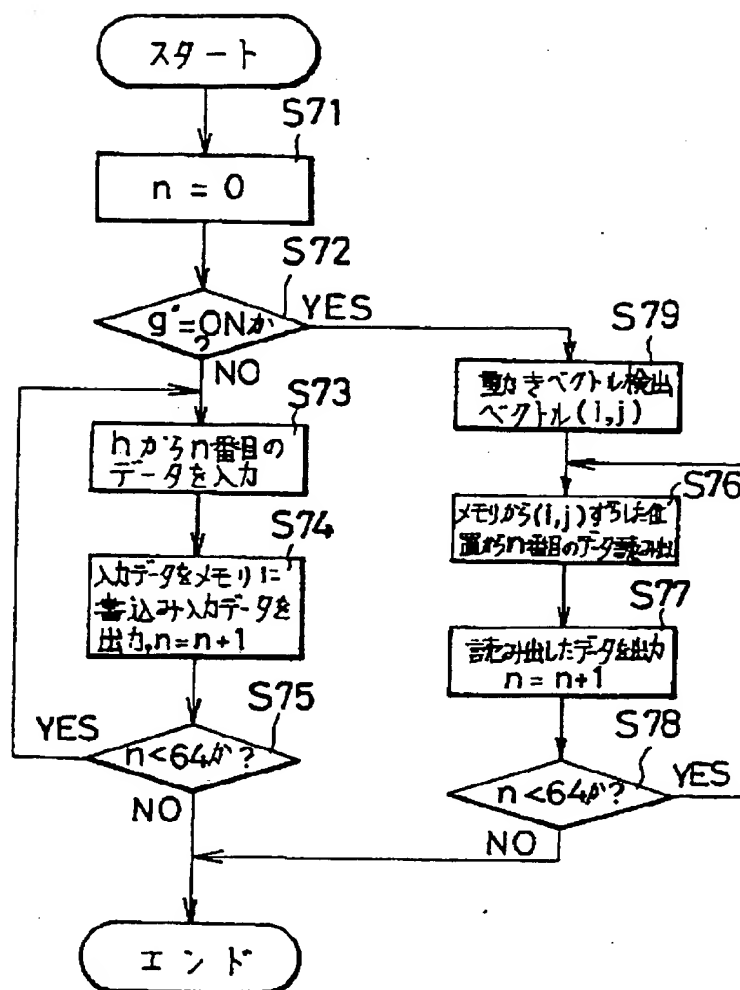
【図7】



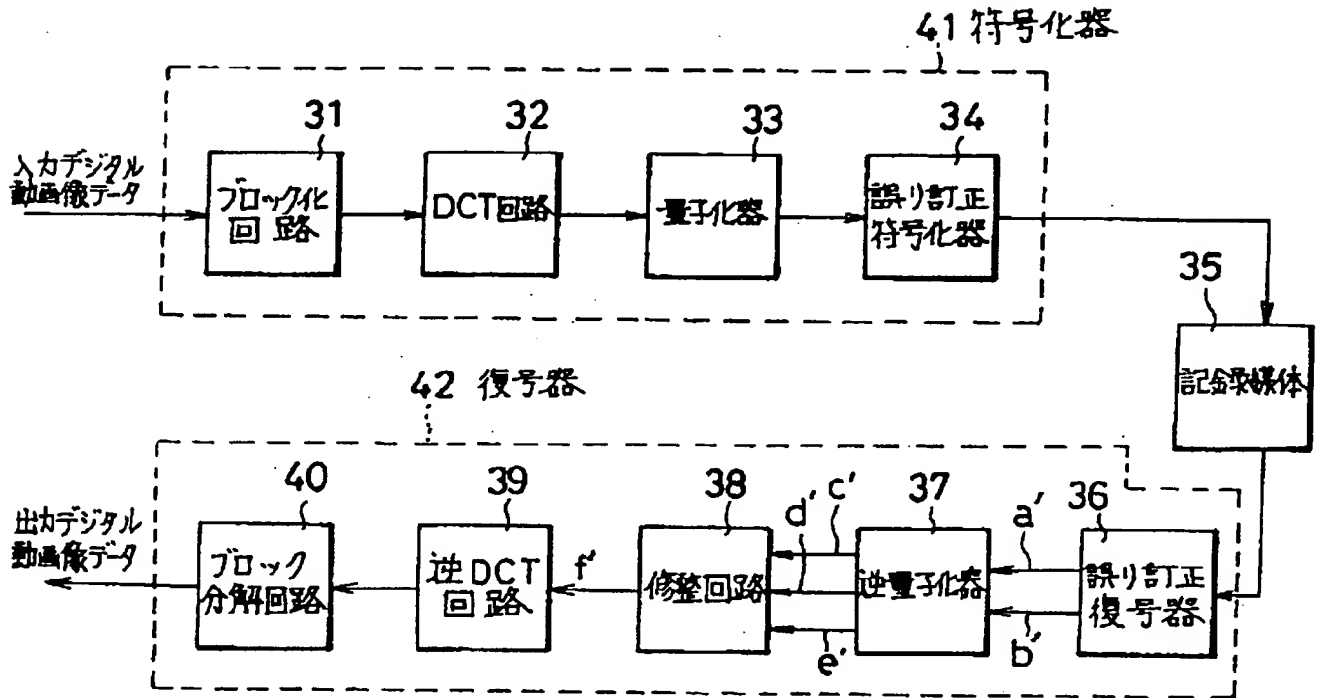
【図13】



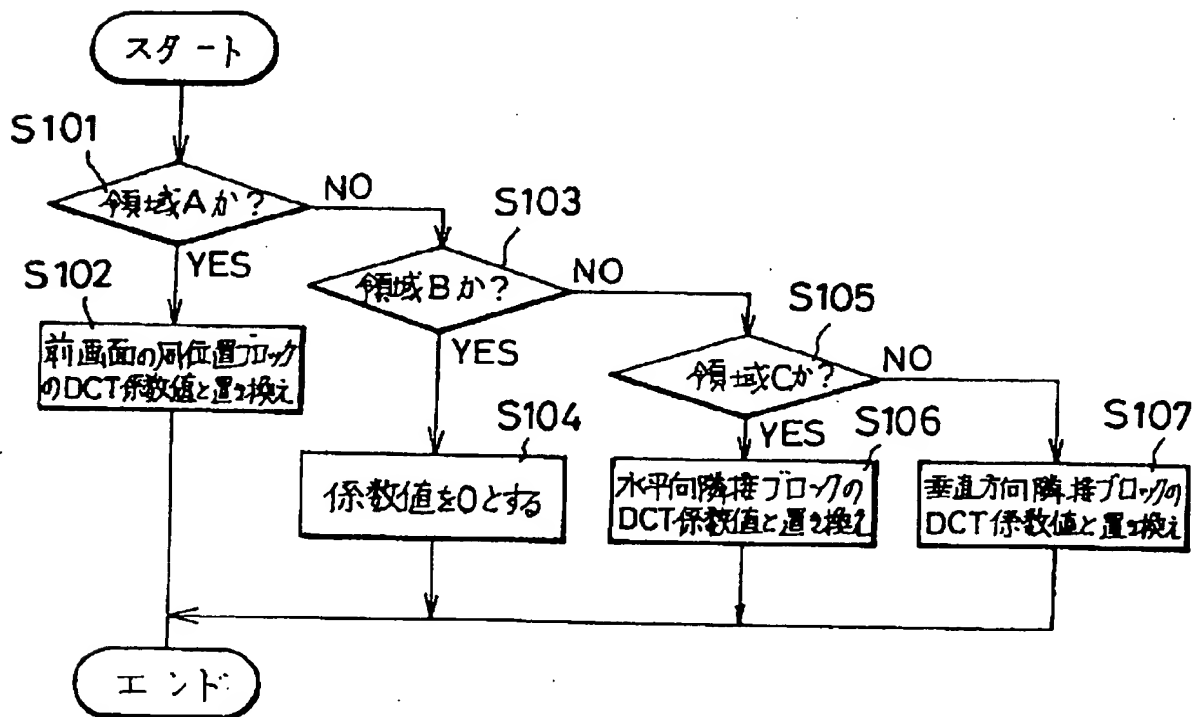
【図8】



【図9】



【図12】



【図11】

